Experiment -1.5

Student Name: HARSH RAJ SINGH **UID:** 22BDO10006
Branch: CSE(DEVOPS) **Section/Group:**22BCD-1/A
Semester: 4TH **Date of Performance:** 20/02/2024
Subject Name:GIT AND GITHUB **Subject Code:** 22CSH-293

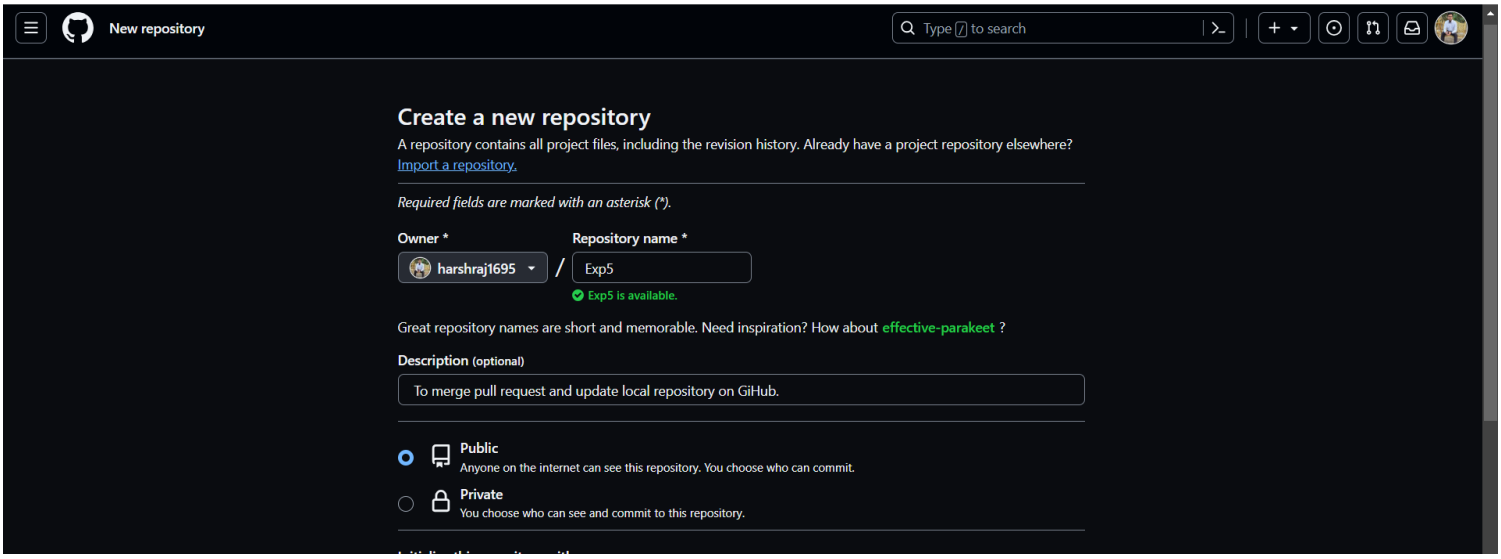
1. Aim/Overview of the practical:. **:** To merge pull request and update local repository on GiHub.

2. Software used: Git Bash and Github.

3. Hardware Used: Computer system.

4. Steps for experiment:

1. Create a file on local device and open that folder with Git Bash.
2. Create a repository on Github with exp5 name and copy the http link.



3. Make clone of the repository using *"git clone http link"*

4. Change the current working directory using **"cd"** command: *"cd exp5"*

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash
$ git clone https://github.com/harshraj1695/Exp5.git
Cloning into 'Exp5'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), 12.79 KiB | 297.00 KiB/s, done.

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash
$ cd Exp5

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$
```

5. Create and edit a file using **"vi filename"** command: *"vi ab.c"*

MINGW64:/d/CU/GIT/gitbash/Exp5

```
#include<stdio.h>
 int main(){
        printf("arsh raj singh");
        return 0;
 }

~
~
~
~
```

6. Add the file using **"git add "** command

7. Commit the file using **"git commit -m"message""** command.

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git add .
warning: in the working copy of 'new.c', LF will be replaced by CRLF the next time Git touches it

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git commit -m "commiting in the main branch"
[main 45e556d] commiting in the main branch
 1 file changed, 6 insertions(+)
 create mode 100644 new.c

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$
```

8. Now Create a branch and move into the branch using “git checkout -b branch_name”

9. Do the changes in the file “new.c”

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git checkout -b nope
Switched to a new branch 'nope'

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$ vi new.c

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$
```

10. Now add and commit the changes of the file.

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$ git commit -am"direct commiting the file without adding in staging area"
warning: in the working copy of 'new.c', LF will be replaced by CRLF the next time Git touches it
[nope 759cad3] direct commiting the file without adding in staging area
 1 file changed, 2 insertions(+), 1 deletion(-)

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$
```

11. Now Shift to the main branch using “git checkout main”.

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (nope)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
```

12. Now shoe the difference between two branches by “dit diff branch1..branch2”

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git diff main..nope
diff --git a/new.c b/new.c
index 107e7b4..59aa2e6 100644
--- a/new.c
+++ b/new.c
@@ -1,6 +1,7 @@
 #include<stdio.h>
  int main(){
-        printf("arsh raj singh");
+        printf("Harsh raj singh");
+        printf("Harsh is a CSE student");
         return 0;
  }
```

12. Now merge the two branch using “git merge branch_name”.

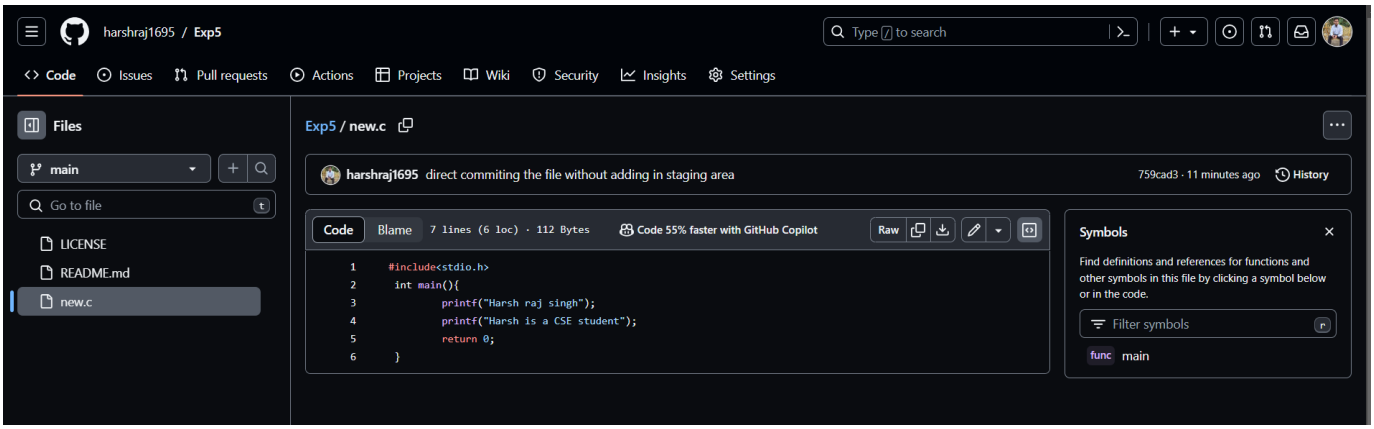
13. After successfully merging the two branch delete the branch using”git branch -d branch_name”.

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git merge nope
Updating 45e556d..759cad3
Fast-forward
 new.c | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git brnach -d nope
git: 'brnach' is not a git command. See 'git --help'.

The most similar command is
        branch

singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git branch -d nope
Deleted branch nope (was 759cad3).
```

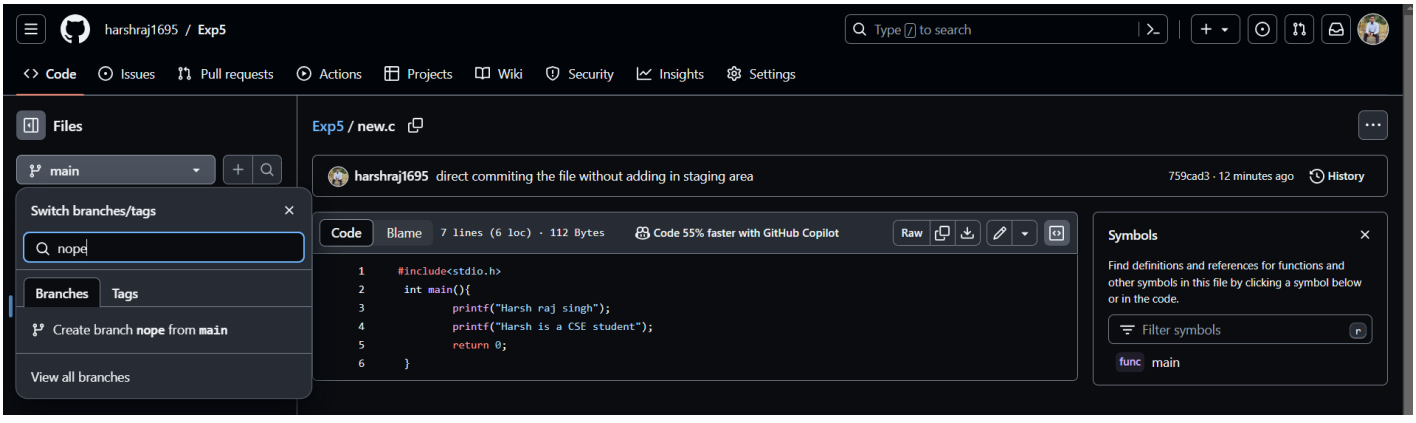
14. Now push the changes in the remote repository using "git push origin main".

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 710 bytes | 710.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/harshraj1695/Exp5.git
   d8d6289..759cad3  main -> main
```

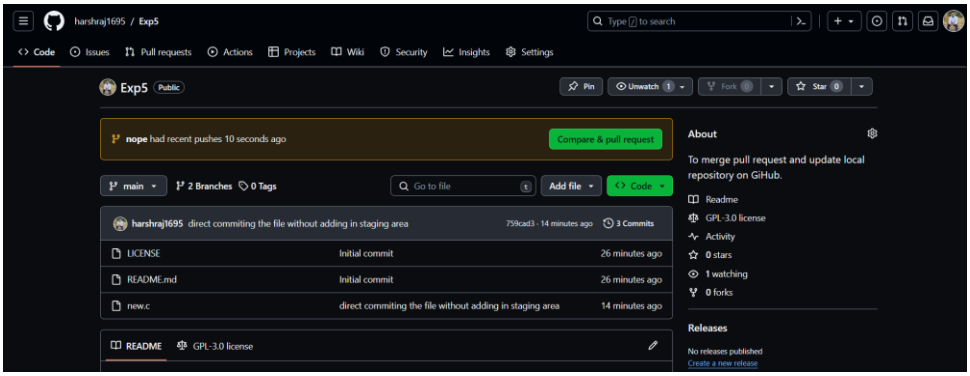
15. Go to the remote repository and verify the changes.



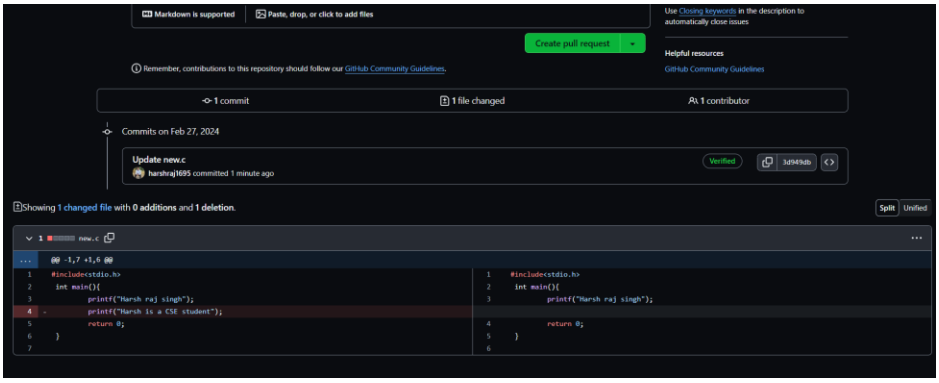
15. Now create a branch at Github and edit the file.



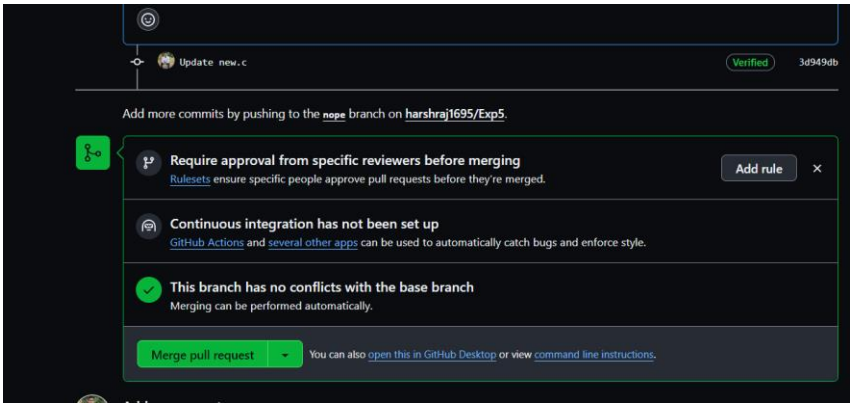
15. Click on “Compare pull request” to create the pull request.



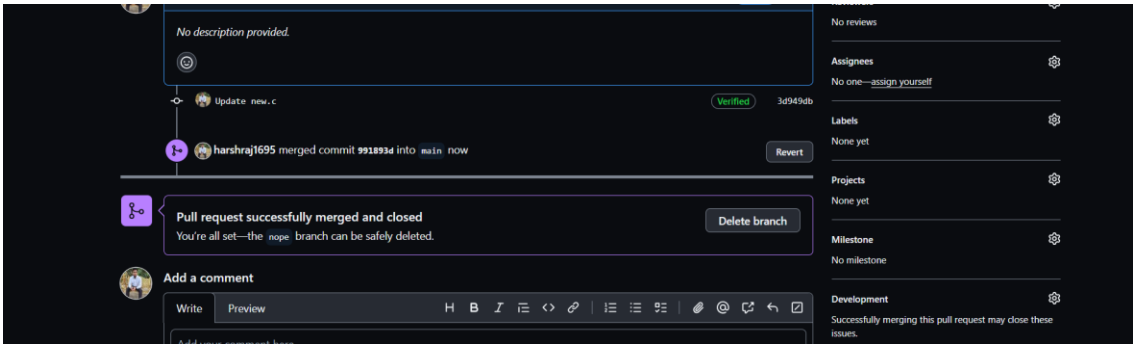
15. Veify the changes and click on “Create pull request”.



15. Click on “merge pull request”.



15. After merging click on “delete branch” to delete the branch.



15. After deleting the branch got to bash and pull the changes by clicking the “git pull origin main”.

```
singh@HARSH1695 MINGW64 /d/CU/GIT/gitbash/Exp5 (main)
$ git pull origin main
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), 1.83 KiB | 234.00 KiB/s, done.
From https://github.com/harshraj1695/Exp5
 * branch            main       -> FETCH_HEAD
   759cad3..991893d  main       -> origin/main
Updating 759cad3..991893d
Fast-forward
 new.c | 1 -
 1 file changed, 1 deletion(-)
```

5. Result/Output/Writing Summary:

In this experiment we have learn about Creating a branch on both local and remote repository and merging the branches bioth on local and remote repository.

Learning outcomes (What I have learnt):

1. Learnt the difference between git merge and pull request.

2. Learnt how to make the merge request on github and git bash.

3. Learnt about editing the files.

4. Learnt how to delete a branch in github and bash.

5. Also learnt about comparing the merged files.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			